Solving Device Tree Issues - part 3

Using devicetree is painful. The framework does not help to develop correct system descriptions (device tree source) and drivers. Errors are often not visible or are hard to understand. But tools and process to make device tree easier to use are being added to Linux. This session will present new tools and techniques that have recently arrived in mainline or are under active development.

This is the third chapter of an ongoing story.

Frank Rowand, Sony

October 13, 2016

161013_0835

Solving Device Tree Issues - part 3

Last year I presented some under-development tools and techniques to debug devicetree issues.

This year I will provide an update on the status of those tools and present some new tools and techniques.

Read this later skip

Any slides with 'skip' in the upper right hand corner will be skipped over in my talk. They contain information that will be useful when the slides are used for reference.

Why this talk?

Debugging device tree problems is not easy.

Why this talk?

Debugging device tree problems is not easy.

At the end of this talk, you will know how to use a new tool to better understand issues related to properties.

Obligatory Outline

Update of Part 1

Device tree concepts

A new feature

A new tool

Chapter 0

Update of Part 1

Update of Part 1

Part 1 slides from elce 2015:

http://elinux.org/images/0/04/Dt_debugging_elce_2015_151006_0421.pdf

Supporting material for Part 1:

http://elinux.org/Device_Tree_frowand

Update of Part 1

dtdiff
  - renamed to scripts/dtc/dtx_diff
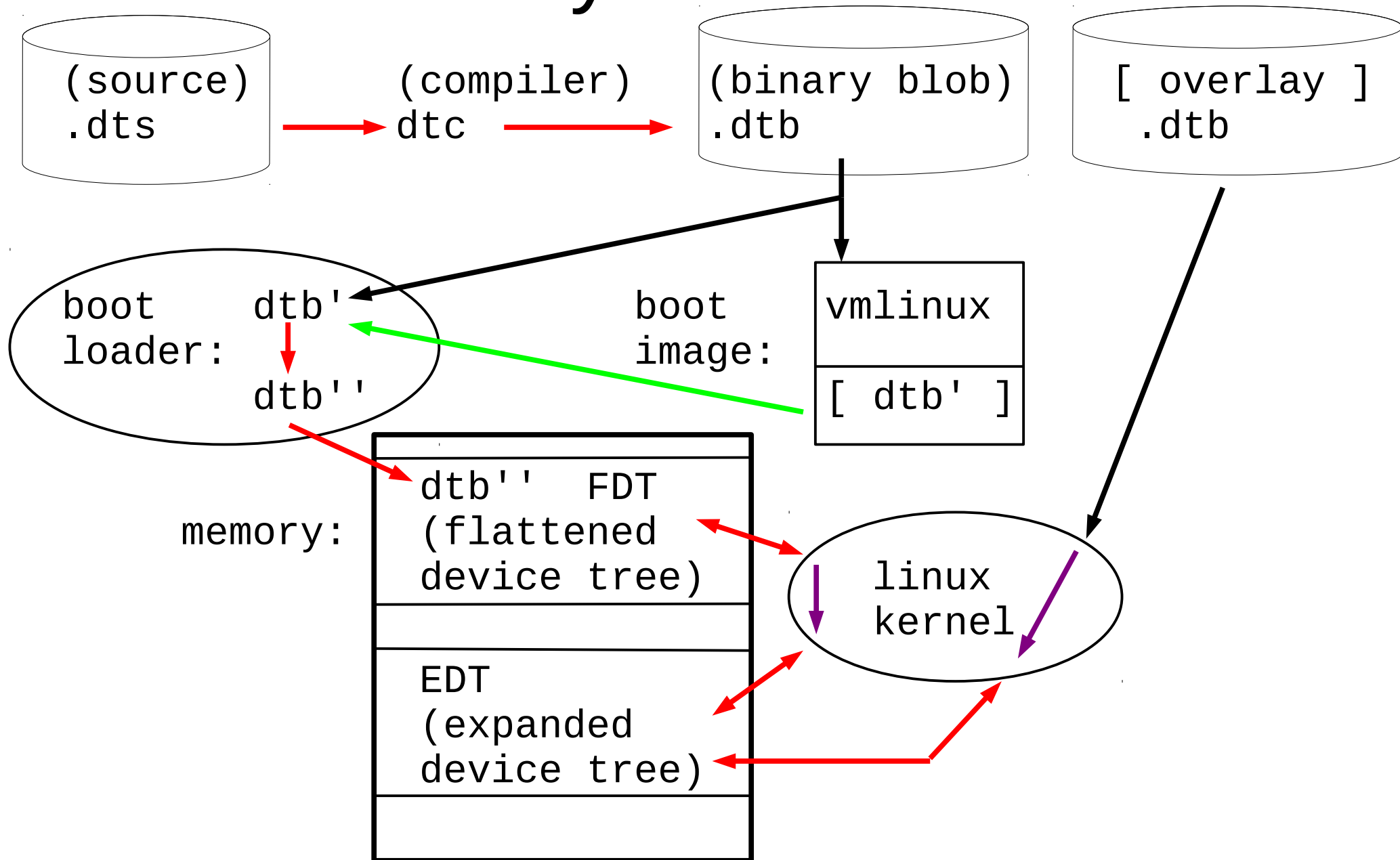 - merged in 4.6-rc1

DT data life cycle

```
device tree source
   .dts  file
   .dtsi files
   .h    files
```

device tree
on booted
target

DT data life cycle

DT data life cycle

dtc creates .dtb from .dts

boot loader copies .dtb into memory FDT

Linux kernel reads FDT, creates Expanded DT

.dtb may be modified by
build process
boot loader

FDT and Expanded DT may be modified by
Linux kernel

dtx_diff - compare two objects

dtx_diff compares device trees in various formats

- source (.dts and the .dtsi includes)

- dtb (binary blob)

- file system tree

dtx_diff - process one .dts

For one source device tree

   - pre-process include file directives and create
   resulting source (that is, converts .dts files
   and included .dtsi files into a single .dts)

Update of Part 1

dt_to_config
   - vastly improved
   - merged in 4.8-rc1

dt_to_config examples and use shown in

   Solving Device Tree Issues - part 2
   LinuxCon Japan 2016
   http://elinux.org/images/5/50/Dt_debugging_part_2.pdf

The kernel configuration problem

Manual process:

device tree node "compatible"
  ==> driver source
    ==> driver object in makefile
      ==> kernel CONFIG option from makefile
        ==> is the CONFIG option enabled?

dt_to_config process

```
scripts/dtc/dt_to_config                                    \
    arch/arm/boot/dts/qcom-apq8074-dragonboard.dts \
    --short-name                                         \
    --config ${KBUILD_OUTPUT}/.config                    \
```

example of one node:

-d-c-----n--F : coincell@2800 : qcom,pm8941-coincell : drivers/misc/qcom-coincell.c : CONFIG_QCOM_COINCELL : n

fields

-d-c-----n--F : coincell@2800 : qcom,pm8941-coincell : drivers/misc/qcom-coincell.c : CONFIG_QCOM_COINCELL : n

-d-c-----n--F   <---- flags

: coincell@2800   <---- node

: qcom,pm8941-coincell   <---- compatible

: drivers/misc/qcom-coincell.c   <---- driver

: CONFIG_QCOM_COINCELL   <---- config option

: n   <---- config value

create kernel .config fragment

```
dt_to_config                                          \
    arch/arm/boot/dts/qcom-apq8074-dragonboard.dts \
    --short-name                                      \
    --config ${KBUILD_OUTPUT}/.config                 \
    --config-format                                   \
    > dragon_config_info
```

```
$ grep "qcom,pm8941-coincell" dragon_config_info
```

-d-c-----n--F : coincell@2800 : qcom,pm8941-coincell : drivers/misc/qcom-coincell.c : CONFIG_QCOM_COINCELL : n
CONFIG_QCOM_COINCELL is not set
CONFIG_QCOM_COINCELL=y

```
    # -d-c-----n--F : coincell@2800 : qcom,pm8941-coincell : .....
    # CONFIG_QCOM_COINCELL is not set
    # CONFIG_QCOM_COINCELL=y
```

Config fragment available for kernel .config

The kernel configuration problem

Manual process:

device tree node "compatible"
   ==> driver source
     ==> driver object in makefile
       ==> kernel CONFIG option from makefile
         ==> is the CONFIG option enabled?

dt_to_config is an aid to fix configuration issues

The result is NOT auto-configuration.

Human decision making is still required.

Update of Part 1

Tools that remain proof of concept:

- dtc --annotate
- dt_node_info
- dt_stat

Chapter 1

Device tree concepts

why device tree?

A device tree describes hardware that can not be located by probing.

what is device tree?

"A device tree is a tree data structure with nodes that describe the devices in a system."

"Each node has property/value pairs that describe the characteristics of the device being represented."

(source: ePAPR v1.1)

Key vocabulary   skip

node
 - the tree structure
 - contain properties and other nodes

property
 - contains zero or more data values providing
   information about a node

.dts - device tree source file

```
/ {     /* incomplete .dts example */
  compatible = "qcom,apq8074-dragonboard";

  soc: soc {
      compatible = "simple-bus";
      intc: interrupt-controller@f9000000 {
          compatible = "qcom,msm-qgic2";
          interrupt-controller;
      };

      console: serial@f991e000 {
          compatible = "qcom,msm-uartdm-v1.4", "qcom,msm-uartdm";
          reg = <0xf991e000 0x1000>;
          interrupts = <0 108 0x0>;
      };
  };
};
```

.dts - Node — a chunk of HW

```
/ {     /* incomplete .dts example */
  compatible = "qcom,apq8074-dragonboard";

  soc: soc {
     compatible = "simple-bus";
     intc: interrupt-controller@f9000000 {
        compatible = "qcom,msm-qgic2";
        interrupt-controller;
     };

     console: serial@f991e000 {
        compatible = "qcom,msm-uartdm-v1.4", "qcom,msm-uartdm";
        reg = <0xf991e000 0x1000>;
        interrupts = <0 108 0x0>;
     };
  };
};
```

.dts - properties and values

```
/ {    /* incomplete .dts example */
   compatible = "qcom,apq8074-dragonboard";

   soc: soc {
       compatible = "simple-bus";
       intc: interrupt-controller@f9000000 {
           compatible = "qcom,msm-qgic2";
           interrupt-controller;
       };

       console: serial@f991e000 {
           compatible = "qcom,msm-uartdm-v1.4", "qcom,msm-uartdm";
           reg = <0xf991e000 0x1000>;
           interrupts = <0 108 0x0>;
       };
   };
};
```

.dts - Reference skip

Thomas Pettazzoni's ELC 2014 talk
"Device Tree For Dummies" is an excellent
introduction to device tree source and concepts.

http://elinux.org/images/f/f9/
  Petazzoni-device-tree-dummies_0.pdf

https://www.youtube.com/watch?v=uzBwHFjJ0vU

More references at

http://elinux.org/Device_Tree_presentations_papers_articles
  "introduction to device tree, overviews, and howtos" section

Chapter 2

A new feature

A new tool

NOT submitted to mainline yet.
See notes at end of talk to get the patches.

Problem Statement

I want to be able to examine how the kernel code uses device tree properties.

Problem Statement

I want to be able to examine how the kernel code uses device tree properties.

I want visibility of when properties are accessed.

What properties were or were not accessed?

Is there a discrepancy between the device tree and the kernel code?

Does the device tree source contain the correct properties?

Kernel Code Perspective

Did my driver or a framework

- read a property value

- attempt to read a property value that did not exist

- not attempt to read a property that exists in the device tree

Device Tree Source Perspective

Does the device tree source contain

- the necessary properties

- properties that should not be present

- properties that are not used by the kernel

Solution

Use printk() to report attempted accesses
of properties.

Solution

WARNING:

Some drivers access the device tree data structures directly instead of using the property access APIs.

Solution

WARNING:

Due to internal framework implementation, not all accesses of properties will be printed.

This feature is extremely useful, but be aware of the few corner cases.

Solution

WARNING:

Due to internal framework implementation, not all accesses of properties will be printed.

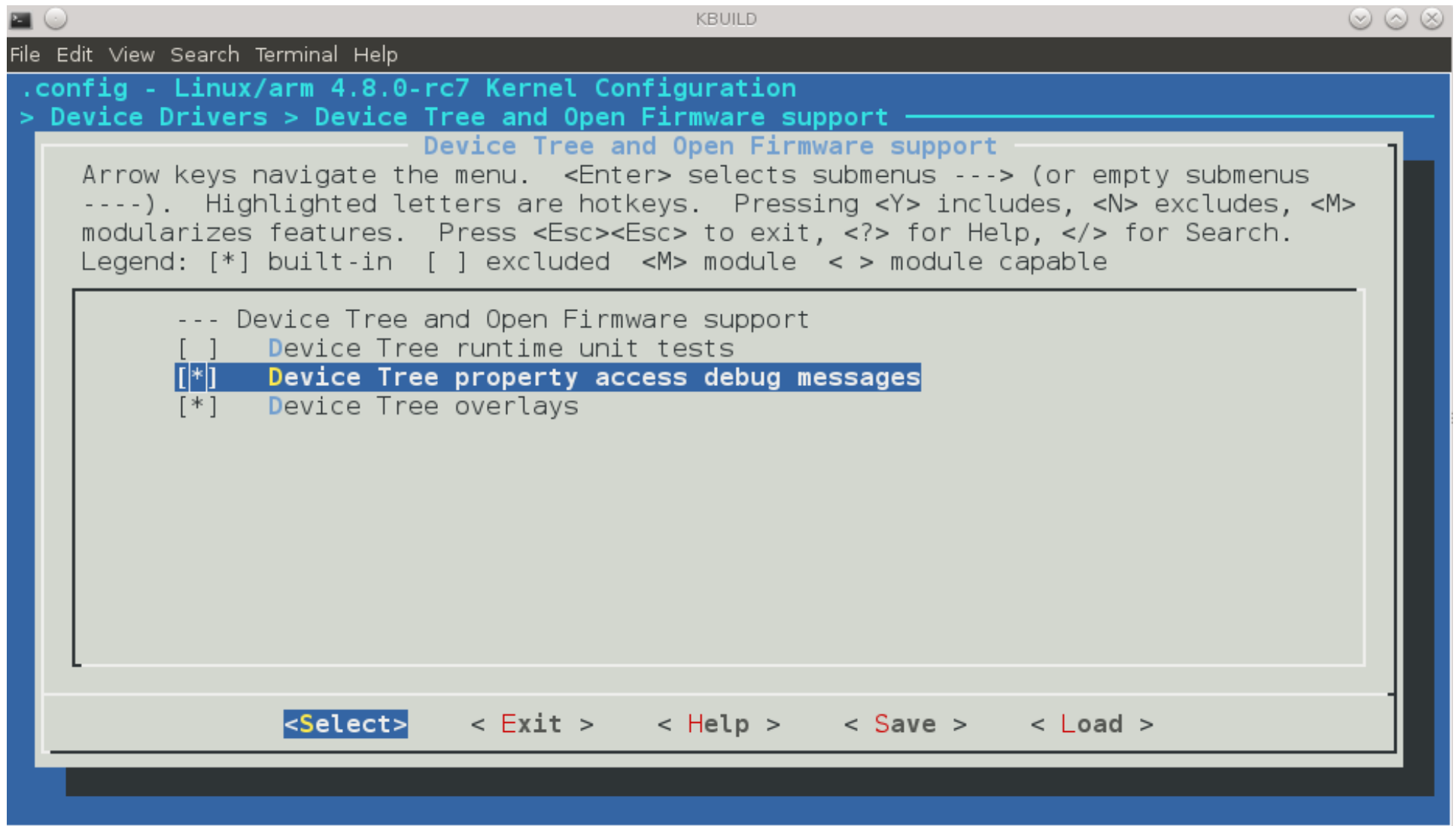This feature is extremely useful, but be aware of the few corner cases.

If you treat this feature as a black box, you will at times be misled. The feature is most useful when used in conjunction with examining and understanding the kernel code.

Solution

WARNING:

See the "caveats" slides near the end of this presentation for some more details.

CONFIG_OF_DEBUG_PROP

Enable Property Access Messages

- Select CONFIG_OF_DEBUG_PROP

- Ensure the printk buffer is large enough (CONFIG_LOG_BUF_SHIFT)

- Add 'debug' to kernel command line

- Rebuild kernel

- Boot

- Choose one of:

    - capture boot console output
    - capture output of 'dmesg' after booting

Boot Console - OF_FND output

Linux version 4.8.0-rc7-dirty ...

...
OF: fdt:OF_FND   1 / NULL 0
OF: fdt: -> unflatten_device_tree()

...
OF: OF_FND   0 / name 1

...
OF: OF_FND   0 /smd/rpm/rpm_requests/pm8941-regulators/5vs2 name 5
OF: OF_FND -22 /smd/rpm/rpm_requests/pm8941-regulators/5vs2 device_type 0
OF: fdt: <- unflatten_device_tree()
OF: fdt:OF_FND   2 / NULL 0
OF: OF_FND   0 /chosen stdout-path 17
OF: adding DT alias:serial0: stem=serial id=0 node=/soc/serial@f991e000
OF: adding DT alias:usid0: stem=usid id=0 node=/soc/spmi@fc4cf000/pm8941@0
OF: adding DT alias:usid4: stem=usid id=4 node=/soc/spmi@fc4cf000/pm8841@4
OF: OF_FND   0 /cpus/cpu@0 reg 4
OF: OF_FND   0 /cpus/cpu@0 enable-method 17
OF: OF_FND   0 /cpus/cpu@1 reg 4
OF: OF_FND   0 /cpus/cpu@2 reg 4
OF: OF_FND   0 /cpus/cpu@3 reg 4

...

Boot Console - ignore unflatten

Linux version 4.8.0-rc7-dirty ...

...
OF: fdt:OF_FND   **1** / NULL 0   <----- '1' is magic value 'OF_FND_IGNORE_BEGIN
OF: fdt: **-> unflatten_device_tree()**

...
OF: OF_FND   0 / name 1

...
OF: OF_FND   0 /smd/rpm/rpm_requests/pm8941-regulators/5vs2 name 5
OF: OF_FND -22 /smd/rpm/rpm_requests/pm8941-regulators/5vs2 device_type 0
OF: fdt: **<- unflatten_device_tree()**
OF: fdt:OF_FND   **2** / NULL 0   <----- '2' is magic value 'OF_FND_IGNORE_END

Properties are accessed during the unflatten
phase. These are not useful accesses for
debugging purposes - they are not driver
or framework accesses.

Boot Console - useful output

...
OF: fdt: <- unflatten_device_tree()
OF: fdt:OF_FND   **2** / NULL 0   <----- '2' is magic value 'OF_FND_IGNORE_END
OF: OF_FND   0 /chosen stdout-path 17
OF: adding DT alias:serial0: stem=serial id=0 node=/soc/serial@f991e000
OF: adding DT alias:usid0: stem=usid id=0 node=/soc/spmi@fc4cf000/pm8941@0
OF: adding DT alias:usid4: stem=usid id=4 node=/soc/spmi@fc4cf000/pm8841@4
OF: OF_FND   0 /cpus/cpu@0 reg 4
OF: OF_FND   0 /cpus/cpu@0 enable-method 17
OF: OF_FND   0 /cpus/cpu@1 reg 4
OF: OF_FND   0 /cpus/cpu@2 reg 4
OF: OF_FND   0 /cpus/cpu@3 reg 4

...

OF_FND - message format

Fields:

- pr_fmt() prefix
- 'OF_FND'
- status
- full node name
- property name
- size of property

```
OF: OF_FND   0 /cpus/cpu@0 reg 4
OF: OF_FND   0 /cpus/cpu@0 enable-method 17
OF: OF_FND   0 /cpus/cpu@1 reg 4
```

OF_FND - status values

Values returned by the property access API

```
  0   success
-22   EINVAL
-61   ENODATA
-75   EOVERFLOW
-84   EILSEQ
```

Values for internal use of OF_FND tools

```
  1   OF_FND_IGNORE_BEGIN
  2   OF_FND_IGNORE_END
```

Using OF_FND messages

OF_FND messages can be manually examined for debugging purposes.

Boot Console - flood of messages

```
$ grep OF_FND console | wc -l
  17963
```

Boot Console - flood of messages

```
$ grep OF_FND console | wc -l
   17963
```

I will reduce the flood of messages to just one node.

This **simple node** can provide some insights in how to use the OF_FND messages.

Boot Console - reduce flood (1)

(1) Reduce flood of messages to just one node.

$ grep OF_FND console | grep coincell | sort -u

```
OF: OF_FND    0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 compatible 21
OF: OF_FND    0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 name 9
OF: OF_FND    0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 status 8
OF: OF_FND -22 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 device_type 0
```

Boot Console - reduce flood (2a)

(2a) Will remove node name path prefix

$ grep OF_FND console | grep coincell | sort -u

```
OF: OF_FND   0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 compatible 21
OF: OF_FND   0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 name 9
OF: OF_FND   0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 status 8
OF: OF_FND -22 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 device_type 0
```

Boot Console - reduce flood (2a)

(2a) Will remove node name path prefix

$ grep OF_FND console | grep coincell | sort -u

```
OF: OF_FND   0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 compatible 21
OF: OF_FND   0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 name 9
OF: OF_FND   0 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 status 8
OF: OF_FND -22 /soc/spmi@fc4cf000/pm8941@0/coincell@2800 device_type 0
```

TROUBLE IN PARADISE:

I failed to remove the messages during unflatten_device_tree(). All of the slides using grep will have some extra properties.

Boot Console - reduced flood (2b)

(2b) Remove node name path prefix

$ grep OF_FND console | grep coincell | sort -u \
  | sed -e 's|\/.*\/||'

```
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0
```

Compare OF_FND to .dts

```
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Compare OF_FND to .dts

```
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Compare OF_FND to .dts

```
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

property "name"

Deprecated, will not appear in the .dtb

If the "name" property occurs in the .dts, the dtc (compiler) will remove it in function check_name_properties().

In the Linux kernel, the unflatten_device_tree() code will create the "name" property from the node name in function populate_properties().

Compare OF_FND to .dts

```
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Compare OF_FND to .dts

```
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0
```

```
coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Compare OF_FND to .dts

```
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0
```

This is an example of a property accessed in unflatten_device_tree().

Summarize - in dts, accessed

```
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0


coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Summarize - in dts, not accessed

```
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0


coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Summarize - not in dts

```
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 status 8
OF: OF_FND -22 coincell@2800 device_type 0
```

```
coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "disable";
};
```

Automation

This simple node was not too difficult to examine, but imagine repeating the process for the entire device tree

(1062 lines for this example device tree)

scripts/dtc/dt_prop

A new tool to automate the comparison of the OF_FND property access messages to the device tree source.

scripts/dtc/dt_prop

```
$ scripts/dtc/dt_prop --help

Usage:

dt_prop TARGET_LOG DTx

    Report differences between properties accessed
    on the target system (reported in TARGET_LOG)
    and properties that are in DTx.
```

dt_prop - (redacted example)

```
$ dt_prop --all-prop        \
  --node-match coincell  \
  console                    \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
        compatible = <>;
-       interrupts;
-       interrupts-extended;
+       qcom,charge-enable;
-       qcom,charger-disable;
```

dt_prop - (redacted example)

Looks suspiciously similar to the output of dtx_diff

dt_prop - algorithm (1)

Looks suspiciously similar to the output of dtx_diff

- Generate dts_target_1
  For each OF_FND message, add a line to
  this device tree source, providing the
  node and property.

dts_target_1

```
/dts-v1/;
/{chosen{stdout-path = <0>;};};
/{cpus{cpu@0{reg = <0>;};};};
/{cpus{cpu@0{enable-method = <0>;};};};
/{cpus{cpu@1{reg = <0>;};};};
/{cpus{cpu@2{reg = <0>;};};};
/{cpus{cpu@3{reg = <0>;};};};
```

dt_prop - algorithm (2)

Looks suspiciously similar to the output of dtx_diff

- Generate dts_target_1
  For each OF_FND message, add a line to
  this device tree source, providing the
  node and property.

- dtx_diff dts_target_1 >dts_target_2

dt_prop - algorithm (3)

Looks suspiciously similar to the output of dtx_diff

- Generate dts_target_1
  For each OF_FND message, add a line to
  this device tree source, providing the
  node and property.

- dtx_diff dts_target_1 >dts_target_2

- dtx_diff DTx   >dts_dtx

dt_prop - algorithm (4)

Looks suspiciously similar to the output of dtx_diff

- Generate dts_target_1
  For each OF_FND message, add a line to
  this device tree source, providing the
  node and property.

- dtx_diff dts_target_1 >dts_target_2

- dtx_diff Dtx   >dts_dtx

- scripts/dtc/dts_diff dts_target_2 dts_dtx

dt_prop - differences only

```
$ dt_prop --node-match coincell  \
                                  \
  console                         \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
+       qcom,charge-enable;
+       qcom,rset-ohms = <>;
+       qcom,vset-millivolts = <>;
+       reg = <>;
 };
```

dt_prop - all properties

```
$ dt_prop --node-match coincell  \
    --all-prop                   \
    console                      \
    arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
         compatible = <>;
+        qcom,charge-enable;
+        qcom,rset-ohms = <>;
+        qcom,vset-millivolts = <>;
+        reg = <>;
         status = <>;
 };
```

dt_prop - in dts, accessed

```
$ dt_prop --node-match coincell  \
    --all-prop                    \
    console                       \
    arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
        compatible = <>;
+       qcom,charge-enable;
+       qcom,rset-ohms = <>;
+       qcom,vset-millivolts = <>;
+       reg = <>;
        status = <>;
 };
```

dt_prop - in dts, not accessed

```
$ dt_prop --node-match coincell  \
   --all-prop                     \
   console                        \
   arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
         compatible = <>;
+        qcom,charge-enable;
+        qcom,rset-ohms = <>;
+        qcom,vset-millivolts = <>;
+        reg = <>;
         status = <>;
 };
```

dt_prop - not in dts (none)

```
$ dt_prop --node-match coincell  \
  --all-prop                      \
  console                         \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
         compatible = <>;
+        qcom,charge-enable;
+        qcom,rset-ohms = <>;
+        qcom,vset-millivolts = <>;
+        reg = <>;
         status = <>;
 };
```

dt_prop - disabled nodes

Property values are redacted, so the value of the status property from dt_prop can not be used to determine if the node is disabled. In this example, the .dts contains: status = "disable";

```
# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
        compatible = <>;
+       qcom,charge-enable;
+       qcom,rset-ohms = <>;
+       qcom,vset-millivolts = <>;
+       reg = <>;
        status = <>;
 };
```

dt_prop - disabled nodes

```
$ dt_prop --node-match coincell  \
  --all-prop --tag-disabled      \
  console                        \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 // *****  node disabled  *****
 coincell@2800 {
         compatible = <>;
+        qcom,charge-enable;
+        qcom,rset-ohms = <>;
+        qcom,vset-millivolts = <>;
+        reg = <>;
         status = <>;
 };
```

Complicate the example

Change status from "disabled" to "ok" to enable the node.

Complicate the example

Change status from "disabled" to "ok" to enable the node.

You should expect the driver probe function to access more properties.

Changes from status = "disabled"

```
$ grep OF_FND console | grep coincell | sort -u \
  | sed -e 's|\/.*\/||'
OF: OF_FND    0 coincell@2800 compatible 21
OF: OF_FND    0 coincell@2800 name 9
OF: OF_FND    0 coincell@2800 qcom,rset-ohms 4
OF: OF_FND    0 coincell@2800 qcom,vset-millivolts 4
OF: OF_FND    0 coincell@2800 reg 4
OF: OF_FND    0 coincell@2800 status 3
OF: OF_FND -22 coincell@2800 assigned-clock-parents 0
OF: OF_FND -22 coincell@2800 assigned-clock-rates 0
OF: OF_FND -22 coincell@2800 device_type 0
OF: OF_FND -22 coincell@2800 dma-coherent 0
OF: OF_FND -22 coincell@2800 interrupts 0
OF: OF_FND -22 coincell@2800 interrupts-extended 0
OF: OF_FND -22 coincell@2800 msi-parent 0
OF: OF_FND -22 coincell@2800 pinctrl-0 0
OF: OF_FND -22 coincell@2800 power-domains 0
OF: OF_FND -22 coincell@2800 qcom,charger-disable 0
OF: OF_FND -22 coincell@2800 reg-names 0
OF: OF_FND -22 coincell@2800 samsung,power-domain 0
```

Compare OF_FND to .dts

```
$ grep OF_FND console | grep coincell | sort -u \
  | sed -e 's|\/.*\/||'
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 qcom,rset-ohms 4
OF: OF_FND   0 coincell@2800 qcom,vset-millivolts 4
OF: OF_FND   0 coincell@2800 reg 4
OF: OF_FND   0 coincell@2800 status 3
OF: OF_FND -22 coincell@2800 assigned-clock-parents 0
OF: OF_FND -22 coincell@2800 assigned-clock-rates 0
OF: OF_FND -22 coincell@2800 device_type 0
OF: OF_FND -22 coincell@2800 dma-coherent 0
OF: OF_FND -22 coincell@2800 interrupts 0
OF: OF_FND -22 coincell@2800 interrupts-extended 0
OF: OF_FND -22 coincell@2800 msi-parent 0
OF: OF_FND -22 coincell@2800 pinctrl-0 0
OF: OF_FND -22 coincell@2800 power-domains 0
OF: OF_FND -22 coincell@2800 qcom,charger-disable 0
OF: OF_FND -22 coincell@2800 reg-names 0
OF: OF_FND -22 coincell@2800 samsung,power-domain 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "ok";
};
```

Summarize - in dts, accessed

```
$ grep OF_FND console | grep coincell | sort -u \
  | sed -e 's|\/.*\/||'
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 qcom,rset-ohms 4
OF: OF_FND   0 coincell@2800 qcom,vset-millivolts 4
OF: OF_FND   0 coincell@2800 reg 4
OF: OF_FND   0 coincell@2800 status 3
OF: OF_FND -22 coincell@2800 assigned-clock-parents 0
OF: OF_FND -22 coincell@2800 assigned-clock-rates 0
OF: OF_FND -22 coincell@2800 device_type 0
OF: OF_FND -22 coincell@2800 dma-coherent 0
OF: OF_FND -22 coincell@2800 interrupts 0
OF: OF_FND -22 coincell@2800 interrupts-extended 0
OF: OF_FND -22 coincell@2800 msi-parent 0
OF: OF_FND -22 coincell@2800 pinctrl-0 0
OF: OF_FND -22 coincell@2800 power-domains 0
OF: OF_FND -22 coincell@2800 qcom,charger-disable 0
OF: OF_FND -22 coincell@2800 reg-names 0
OF: OF_FND -22 coincell@2800 samsung,power-domain 0


coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "ok";
};
```

Summarize - in dts, not accessed

```
$ grep OF_FND console | grep coincell | sort -u \
  | sed -e 's|\/.*\/||'
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 qcom,rset-ohms 4
OF: OF_FND   0 coincell@2800 qcom,vset-millivolts 4
OF: OF_FND   0 coincell@2800 reg 4
OF: OF_FND   0 coincell@2800 status 3
OF: OF_FND -22 coincell@2800 assigned-clock-parents 0
OF: OF_FND -22 coincell@2800 assigned-clock-rates 0
OF: OF_FND -22 coincell@2800 device_type 0
OF: OF_FND -22 coincell@2800 dma-coherent 0
OF: OF_FND -22 coincell@2800 interrupts 0
OF: OF_FND -22 coincell@2800 interrupts-extended 0
OF: OF_FND -22 coincell@2800 msi-parent 0
OF: OF_FND -22 coincell@2800 pinctrl-0 0
OF: OF_FND -22 coincell@2800 power-domains 0
OF: OF_FND -22 coincell@2800 qcom,charger-disable 0
OF: OF_FND -22 coincell@2800 reg-names 0
OF: OF_FND -22 coincell@2800 samsung,power-domain 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "ok";
};
```

Summarize - not in dts

```
$ grep OF_FND console | grep coincell | sort -u \
  | sed -e 's|\/.*\/||'
OF: OF_FND   0 coincell@2800 compatible 21
OF: OF_FND   0 coincell@2800 name 9
OF: OF_FND   0 coincell@2800 qcom,rset-ohms 4
OF: OF_FND   0 coincell@2800 qcom,vset-millivolts 4
OF: OF_FND   0 coincell@2800 reg 4
OF: OF_FND   0 coincell@2800 status 3
OF: OF_FND -22 coincell@2800 assigned-clock-parents 0
OF: OF_FND -22 coincell@2800 assigned-clock-rates 0
OF: OF_FND -22 coincell@2800 device_type 0
OF: OF_FND -22 coincell@2800 dma-coherent 0
OF: OF_FND -22 coincell@2800 interrupts 0
OF: OF_FND -22 coincell@2800 interrupts-extended 0
OF: OF_FND -22 coincell@2800 msi-parent 0
OF: OF_FND -22 coincell@2800 pinctrl-0 0
OF: OF_FND -22 coincell@2800 power-domains 0
OF: OF_FND -22 coincell@2800 qcom,charger-disable 0
OF: OF_FND -22 coincell@2800 reg-names 0
OF: OF_FND -22 coincell@2800 samsung,power-domain 0

coincell@2800 {
        compatible = "qcom,pm8941-coincell";
        qcom,charge-enable;
        qcom,rset-ohms = <0x834>;
        qcom,vset-millivolts = <0xbb8>;
        reg = <0x2800>;
        status = "ok";
};
```

dt_prop - differences only

```
$ dt_prop --node-match coincell  \
                                 \
  console                        \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
-        assigned-clock-parents;
-        assigned-clock-rates;
-        dma-coherent;
-        interrupts;
-        interrupts-extended;
-        msi-parent;
-        pinctrl-0;
-        power-domains;
+        qcom,charge-enable;
-        qcom,charger-disable;
-        reg-names;
-        samsung,power-domain;
 };
```

dt_prop - all properties

```
$ dt_prop --node-match coincell  \
  --all-prop                     \
  console                        \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

 coincell@2800 {
-        assigned-clock-parents;
-        assigned-clock-rates;
         compatible = <>;
-        dma-coherent;
-        interrupts;
-        interrupts-extended;
-        msi-parent;
-        pinctrl-0;
-        power-domains;
+        qcom,charge-enable;
-        qcom,charger-disable;
         qcom,rset-ohms = <>;
         qcom,vset-millivolts = <>;
         reg = <>;
-        reg-names;
-        samsung,power-domain;
         status = <>;
 };
```

dt_prop - in dts, accessed

```
$ dt_prop --node-match coincell  \
  --all-prop                      \
  console                         \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

 coincell@2800 {
-        assigned-clock-parents;
-        assigned-clock-rates;
         compatible = <>;
-        dma-coherent;
-        interrupts;
-        interrupts-extended;
-        msi-parent;
-        pinctrl-0;
-        power-domains;
+        qcom,charge-enable;
-        qcom,charger-disable;
         qcom,rset-ohms = <>;
         qcom,vset-millivolts = <>;
         reg = <>;
-        reg-names;
-        samsung,power-domain;
         status = <>;
 };
```

dt_prop - in dts, not accessed

```
$ dt_prop --node-match coincell  \
  --all-prop                      \
  console                         \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

 coincell@2800 {
-         assigned-clock-parents;
-         assigned-clock-rates;
          compatible = <>;
-         dma-coherent;
-         interrupts;
-         interrupts-extended;
-         msi-parent;
-         pinctrl-0;
-         power-domains;
+         qcom,charge-enable;
-         qcom,charger-disable;
          qcom,rset-ohms = <>;
          qcom,vset-millivolts = <>;
          reg = <>;
-         reg-names;
-         samsung,power-domain;
          status = <>;
 };
```

dt_prop - not in dts

```
$ dt_prop --node-match coincell  \
  --all-prop                     \
  console                        \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

 coincell@2800 {
-         assigned-clock-parents;
-         assigned-clock-rates;
          compatible = <>;
-         dma-coherent;
-         interrupts;
-         interrupts-extended;
-         msi-parent;
-         pinctrl-0;
-         power-domains;
+         qcom,charge-enable;
-         qcom,charger-disable;
          qcom,rset-ohms = <>;
          qcom,vset-millivolts = <>;
          reg = <>;
-         reg-names;
-         samsung,power-domain;
          status = <>;
 };
```

Driver - source

What property accesses should we expect from reading the driver?

Driver - source

```
# drivers/misc/qcom-coincell.c:

qcom_coincell_probe()
    of_property_read_u32(, "reg", )
    of_property_read_bool(, "qcom,charger-disable")
    of_property_read_u32(, "qcom,rset-ohms", )
    of_property_read_u32(, "qcom,vset-millivolts", )
```

accessed by driver

```
 coincell@2800 {
        ...
        compatible = <>;
        ...
+       qcom,charge-enable;
-       qcom,charger-disable;
        qcom,rset-ohms = <>;
        qcom,vset-millivolts = <>;
        reg = <>;
        ...
        status = <>;
 };
```

not accessed by kernel

```
 coincell@2800 {
        ...
        compatible = <>;
        ...
+       qcom,charge-enable;
-       qcom,charger-disable;
        qcom,rset-ohms = <>;
        qcom,vset-millivolts = <>;
        reg = <>;
        ...
        status = <>;
 };
```

not accessed by kernel

BUG!

```
coincell@2800 {
        ...
        compatible = <>;
        ...
+       qcom,charge-enable;
-       qcom,charger-disable;
        qcom,rset-ohms = <>;
        qcom,vset-millivolts = <>;
        reg = <>;
        ...
        status = <>;
};
```

I carried forward an old patch that added the
coincell node to the .dtsi. The old patch had
a property for a previous version of the driver.

Sidetrack: something odd   skip

```
$ dt_prop --node-match coincell  \
  console                        \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
-        assigned-clock-parents;
-        assigned-clock-rates;
-        dma-coherent;
-        interrupts;
-        interrupts-extended;
-        msi-parent;
-        pinctrl-0;
-        power-domains;
+        qcom,charge-enable;
-        qcom,charger-disable;
-        reg-names;
-        samsung,power-domain;
 };
```

Sidetrack: something odd... skip

```
# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 coincell@2800 {
-        samsung,power-domain;
 };
```

A Samsung related property access attempted
for a device that is a Qualcomm SOC device.

samsung,power-domain skip

binding document:

    bindings/display/exynos/exynos7-decon.txt

Device-Tree bindings for
Samsung Exynos7 SoC display controller (DECON)

samsung,power-domain

Code fragment:

```
drivers/base/power/domain.c:

genpd_dev_pm_attach()
    ret = of_parse_phandle_with_args(, "power-domains", )
    if (ret < 0) {
        /*
         * Try legacy Samsung-specific bindings
         * (for backwards compatibility of DT ABI)
         */
        of_parse_phandle(, "samsung,power-domain", )
```

samsung,power-domain

In this case, accessing the extraneous property causes no harm.

But this example shows how an otherwise unexpected use of a property can be detected.

Next 2 slides

The next 2 slides were not created by any tool.

They each are the output of two dt_prop runs, pasted together by hand.

node disabled vs. enabled   skip

dt_prop: differences only

```
// *****  node disabled  *****   |
coincell@2800 {                  |   coincell@2800 {
                                 | -        assigned-clock-parents;
                                 | -        assigned-clock-rates;
                                 | -        dma-coherent;
                                 | -        interrupts;
                                 | -        interrupts-extended;
                                 | -        msi-parent;
                                 | -        pinctrl-0;
                                 | -        power-domains;
+        qcom,charge-enable;     | +        qcom,charge-enable;
                                 | -        qcom,charger-disable;
+        qcom,rset-ohms = <>;     |
+        qcom,vset-millivolts = <>;  |
+        reg = <>;               |
                                 |
                                 | -        reg-names;
                                 | -        samsung,power-domain;
};                               |   };
```

node disabled vs. enabled skip

dt_prop: all properties

```
// *****  node disabled  *****   |
 coincell@2800 {                 |     coincell@2800 {
                                 |   -         assigned-clock-parents;
                                 |   -         assigned-clock-rates;
         compatible = <>;        |             compatible = <>;
                                 |   -         dma-coherent;
                                 |   -         interrupts;
                                 |   -         interrupts-extended;
                                 |   -         msi-parent;
                                 |   -         pinctrl-0;
                                 |   -         power-domains;
 +       qcom,charge-enable;     |   +         qcom,charge-enable;
                                 |   -         qcom,charger-disable;
 +       qcom,rset-ohms = <>;    |             qcom,rset-ohms = <>;
 +       qcom,vset-millivolts = <>; |          qcom,vset-millivolts = <>;
 +       reg = <>;               |             reg = <>;
                                 |   -         reg-names;
                                 |   -         samsung,power-domain;
         status = <>;            |             status = <>;
 };                              |     };
```

node name ambiguity - problemskip

```
$ dt_prop --node-match master-kernel  \
                                       \
   console                             \
   arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 master-kernel {
+        #qcom,smem-state-cells = <>;
-        compatible;
+        qcom,entry-name = <>;
 };

 master-kernel {
+        #qcom,smem-state-cells = <>;
-        compatible;
+        qcom,entry-name = <>;
 };
```

node name ambiguity - solutionskip

```
$ dt_prop --node-match master-kernel  \
   --full-path                        \
   console                            \
   arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 /smp2p-modem/master-kernel {
+        #qcom,smem-state-cells = <>;
-        compatible;
+        qcom,entry-name = <>;
 };


 /smp2p-wcnss/master-kernel {
+        #qcom,smem-state-cells = <>;
-        compatible;
+        qcom,entry-name = <>;
 };
```

Caveats

Caveats - Black Box

WARNING:

Due to internal framework implementation, not all accesses of properties will be printed.

This feature is extremely useful, but be aware of the few corner cases.

If you treat this feature as a black box, you will at times be misled. The feature is most useful when used in conjunction with examining and understanding the kernel code.

Caveats- Black Box

Is this a problem?

```
# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
 /smd/modem {
-       compatible;
-       interrupt-parent;
        interrupts = <>;
-       interrupts-extended;
        qcom,ipc = <>;
-       qcom,remote-pid;
        qcom,smd-edge = <>;
-       reg;
-       status;
 };
```

Caveats- Black Box

```
/ {

    ...
    cpus {
        ...
    };
    ...
    smd {
        modem {
            interrupts = <0x0 0x19 0x1>;
            qcom,ipc = <0x10 0x8 0xc>;
            qcom,smd-edge = <0x0>;
        };
    };
```

Caveats- Black Box

Is this a problem? What does the boot log show?

```
OF: OF_FND    0 /smd/modem interrupts 12
OF: OF_FND  -22 /smd/modem interrupt-parent 0
```

Caveats- Black Box

Is this a problem? What does the boot log show?

```
OF: OF_FND   0 /smd/modem interrupts 12
OF: OF_FND -22 /smd/modem interrupt-parent 0
OF: OF_FND -22 /smd #interrupt-cells 0
OF: OF_FND -22 /smd interrupt-parent 0
```

Caveats- Black Box

Is this a problem? What does the boot log show?

```
OF: OF_FND    0 /smd/modem interrupts 12
OF: OF_FND -22 /smd/modem interrupt-parent 0
OF: OF_FND -22 /smd #interrupt-cells 0
OF: OF_FND -22 /smd interrupt-parent 0
OF: OF_FND -22 / #interrupt-cells 0
OF: OF_FND    0 / interrupt-parent 4
```

Caveats- Black Box skip

Is this a problem? What does the boot log show?

```
OF: OF_FND    0 /smd/modem interrupts 12
OF: OF_FND -22 /smd/modem interrupt-parent 0
OF: OF_FND -22 /smd #interrupt-cells 0
OF: OF_FND -22 /smd interrupt-parent 0
OF: OF_FND -22 / #interrupt-cells 0
OF: OF_FND    0 / interrupt-parent 4
```

These lines appear as consecutive lines
in the boot log.

Caveats- Black Box

```
/ {
    interrupt-parent = <0x1>; //inherited by descendants
    ...
    cpus {
        ...
    };
    ...
    smd {
        modem {
            interrupts = <0x0 0x19 0x1>;
            qcom,ipc = <0x10 0x8 0xc>;
            qcom,smd-edge = <0x0>;
        };
    };
    soc: soc {
        intc: interrupt-controller@f9000000 {
            interrupt-controller;
            phandle = <0x1>;
```

Caveats- Black Box

Is this a problem? What does dt_prop show?

Caveats- Black Box skip

Is this a problem? What does dt_prop show?

Walk up the tree with dt_prop:

```
$ dt_prop --all-prop --full-path --node-exact modem \
  console arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

$ dt_prop --all-prop --full-path --node-exact smd \
  console arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

$ dt_prop --all-prop --full-path --node-exact / \
  console arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
```

dt_prop - /smd/modem   skip

```
/smd/modem {
-        compatible;
-        interrupt-parent;
         interrupts = <>;
-        interrupts-extended;
         qcom,ipc = <>;
-        qcom,remote-pid;
         qcom,smd-edge = <>;
-        reg;
-        status;
};
```

dt_prop - /smd/ skip

```
/smd {
-        #address-cells;
-        #interrupt-cells;
-        #size-cells;
-        assigned-clock-parents;
-        assigned-clock-rates;
         compatible = <>;
-        dma-coherent;
-        interrupt-parent;
-        interrupts;
-        interrupts-extended;
-        msi-parent;
-        pinctrl-0;
-        power-domains;
-        reg;
-        samsung,power-domain;
-        status;
```

dt_prop - / skip

```
 / {
          #address-cells = <>;
-         #interrupt-cells;
          #size-cells = <>;
          compatible = <>;
-         dma-coherent;
-         dma-ranges;
          interrupt-parent = <>;
+         model = <>;
-         serial-number;
 };
```

Caveats

Enabling the OF_FND messages increases boot time significantly.

Example, dragonboard with serial console

without OF_FND:

285 lines to console

3 seconds to shell prompt

with OF_FND:

18437 lines to console

120 seconds to shell prompt

Caveats

dt_prop does not check the OF_FND status value

If there is an OF_FND message for a property and the property appears in the device tree then the first column of the dt_prop report will contain a space, even if the access failed (for example, due to an incorrect data length).

The space does NOT mean that a value was read from the property.

Caveats skip

Data values are redacted

Labels are redacted

Some properties are redacted

    - linux,phandle

Caveats - special properties   skip

Some properties are not stored in the kernel device tree as properties. This is due to framework internal implementation details.

These properties are instead stored in fields in device tree structures.

Accessing these fields is sometimes open coded and thus sometimes will not generate an OF_FND message.

Caveats - special properties

The current list of special properties is:
- device_type
- linux,phandle
- name
- phandle

More Info

For some caveats and more info about dt_prop:

```
$ scripts/dtc/dt_prop --help
```

Other Uses of OF_FND Data skip

Context of Property Access skip

Tools and methods to reduce the large number of property access debug messages, but lose information about when the properties are accessed

- grep OF_FND console | sort -u

- dt_prop

Back to the base feature skip

CONFIG_OF_DEBUG_PROP

Uses printk() to report attempted accesses of properties.

Back to the base feature

CONFIG_OF_DEBUG_PROP

Uses printk() to report attempted accesses of properties.

Bonus:

If other debug printk messages are enabled, can view property accesses within the context of the other existing debug messages.

Future Possibilities

If the bindings documents validation project
is completed, then the bindings documents
will be easily machine readable.

If the bindings documents are machine readable,
then dt_prop could be enhanced to indicate
whether the properties in the report are
required or optional.

Trivia - kernel config option   skip

Why add a kernel config option to enable the OF_FND console messages?

1) The macro that prints the messages is gated by the config option. The volume of OF_FND messages is excessive when they are not needed.

2) Dynamic debug messages are enabled too late in the boot process to include many of the property accesses. The config option is used to statically enable the OF_FND messages.

Review - Updates

Compare device trees (source, binary, /sys tree)

- dtx_diff

Investigate and resolve kernel configuration for a given device tree

- dt_to_config

Review - New

Analyze issues related to device tree properties

- CONFIG_OF_DEBUG_PROP

- dt_prop

The growing set of tools

- are individually useful

- can be used as building blocks to create
  more powerful tools

Resources skip

http://elinux.org/Device_Tree_presentations_papers_articles

http://elinux.org/Device_Tree_Reference

devicetree: Kernel Internals and Practical Troubleshooting
Frank Rowand, ELCE 2014

http://elinux.org/ELC_Europe_2014_Presentations

Resources skip

Solving Device Tree Issues - Part 1:
   http://elinux.org/images/0/04/Dt_debugging_elce_2015_151006_0421.pdf

Supporting material for: Solving Device Tree Issues - Part 1:
   http://elinux.org/Device_Tree_frowand

Solving Device Tree Issues - Part 2:

   Frank Rowand, LinuxCon Japan 2016

   http://elinux.org/images/5/50/Dt_debugging_part_2.pdf

Supporting material for: Solving Device Tree Issues - Part 3:
   kernel patches
   scripts/dtc/dts_diff
   scripts/dtc/dt_prop
   http://elinux.org/Device_Tree_frowand

THE END

Thank you for your attention...

Questions?

How to get a copy of the slides

1) leave a business card with me

2) frank.rowand@am.sony.com

3) http://elinux.org/Device_Tree_presentations_papers_articles

4) http://events.linuxfoundation.org